INTERNATIONAL STANDARD

ISO/IEC 23002-4

Second edition
2014-04-15

Information technology — MPEG video technologies —

Part 4: Video tool library

Technologies de l'information — Technologies vidéo MPEG —

Partie 4: Bibliothèque d'outils vidéo

Reference number
ISO/IEC 23002-4:2014(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 23002-4 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This second edition cancels and replaces the first edition (ISO/IEC 23002-4:2010), which has been technically revised. It also incorporates the Amendment ISO/IEC 23002-4:2010/Amd.1:2011.

ISO/IEC 23002 consists of the following parts, under the general title *Information technology — MPEG video technologies*:

— *Part 1: Accuracy requirements for implementation of integer-output 8×8 inverse discrete cosine transform*

— *Part 2: Fixed-point 8×8 inverse discrete cosine transform and discrete cosine transform*

— *Part 3: Representation of auxiliary video and supplemental information*

— *Part 4: Video tool library*

— *Part 5: Reconfigurable media coding conformance and reference software*

Introduction

This part of ISO/IEC 23002 defines the MPEG video tool library, which contains tools drawn from existing MPEG coding standards, such as ISO/IEC 14496-2 and ISO/IEC 14496-10, and ISO/IEC 23001-4 defines the methods capable of describing codec configurations in the reconfigurable video coding (RVC) framework.

This part of ISO/IEC 23002 primarily addresses reconfigurable video aspects and will only focus on the description of representation of video codec configurations under the RVC framework, but could be extended to a more generic reconfigurable media coding (RMC) framework.

The objective of RVC is to offer a framework that is capable of configuring and specifying video codecs as a collection of "higher level" modules by using video coding tools. The video coding tools are defined in video tool libraries. This part of ISO/IEC 23002 defines the MPEG video tool library. The RVC framework principle could also support non-MPEG tool libraries, provided that their developers have taken care to obey the appropriate rules of operation.

For the purpose of framework deployment, an appropriate description is needed to describe configurations of decoders composed of or instantiated from a subset of video tools from either one or more libraries. As illustrated in Figure 1, the configuration information consists of

— bitstream syntax description, and

— network of functional units (FUs) description (also referred to as the decoder configuration)

that together constitute the entire decoder description.

Bitstreams of existing MPEG standards are specified by specific syntax structures and decoders are composed of various coding tools. Therefore, RVC includes support for bitstream syntax descriptions as well as video coding tools. As depicted in Figure 1, a typical RVC decoder requires two types of information, namely the decoder description and the encoded media (e.g. video bitstreams) data.

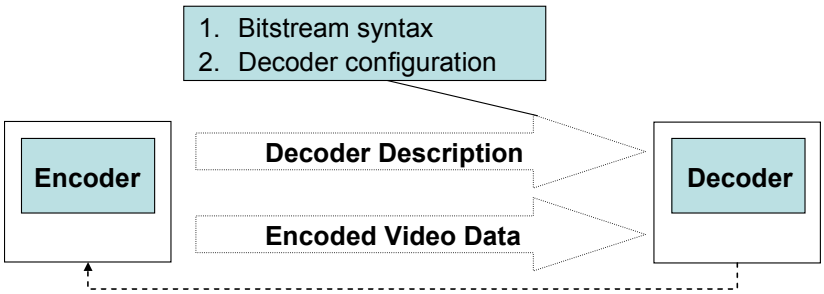

Figure 1 — Conceptual diagram of RVC

A more detailed description of the RVC decoder is illustrated in Figure 2. As shown in Figure 2, the decoder description is required for the configuration of a RVC decoder. The Bitstream Syntax Description (BSD) and FU Network Description (FND) (which compose the Decoder Description) are used to configure or compose an abstract decoder model (ADM) which is instantiated through the selection of FUs from tool libraries optionally with proper parameter assignment. Such ADM constitutes the behavioral reference model used in setting up a decoding solution under the RVC framework. The process of yielding a decoding solution may vary depending on the technologies used for the desired implementations. Examples of the instantiation of an ADM and generation of proprietary decoding solutions can be found in ISO/IEC 23001-4.

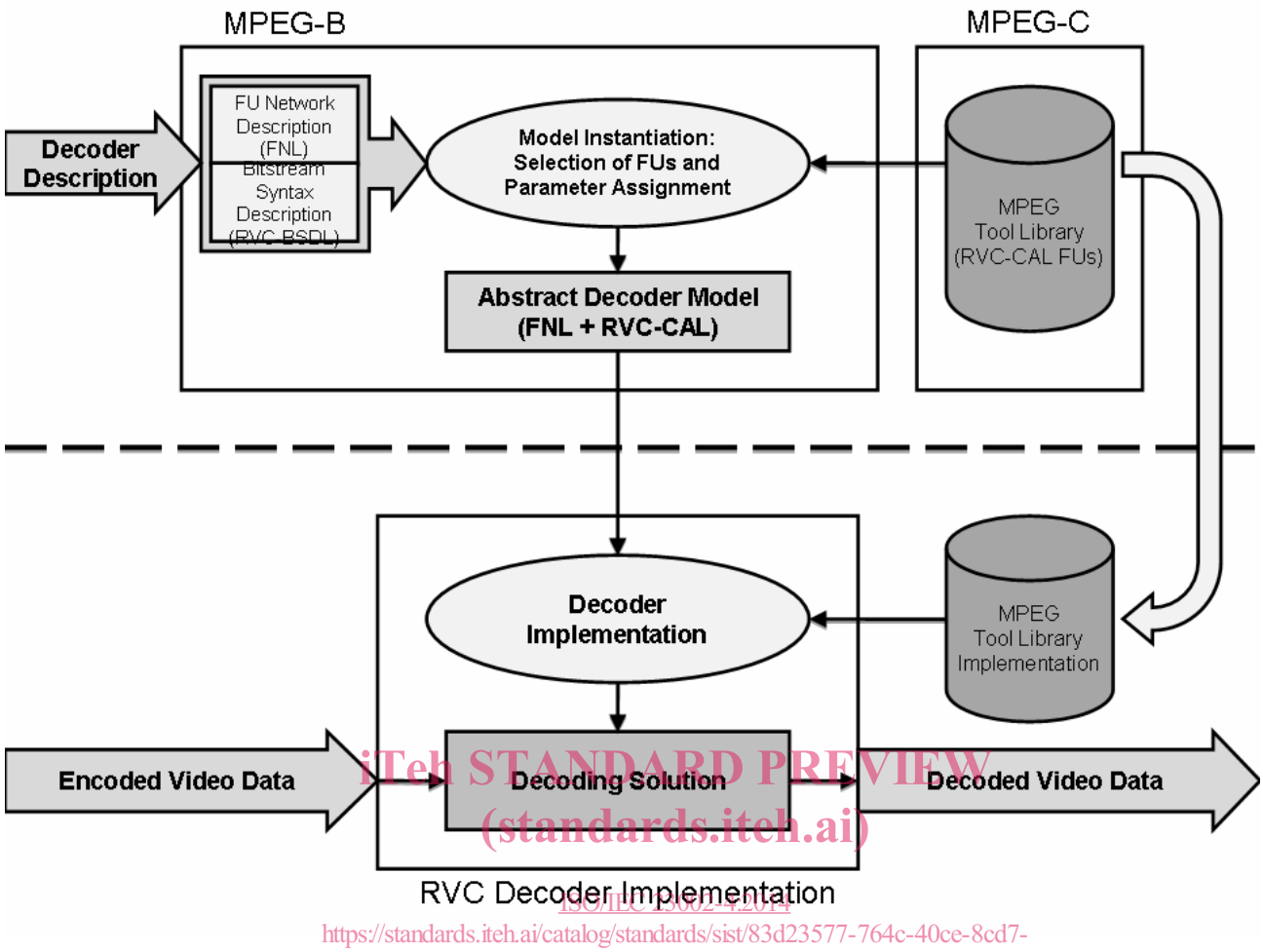

Figure 2 — Graphical representation of the process for setting up a decoding solution under the RVC framework

Within the RVC framework, the decoder description describes a particular decoder configuration and consists of the FND and the BSD. The FND describes the connectivity of the network of FUs used to form a decoder whereas the parsing process for the bitstream syntax is implicitly described by the BSD. These two descriptions are specified using two standard XML-based languages or dialects:

— Functional unit network language (FNL) is a language that describes the FND, known also as “network of FUs”. The FNL specified normatively within the scope of the RVC framework is provided in ISO/IEC 23001-4.

— Bitstream syntax description language (BSDL), standardized in ISO/IEC 23001-5 (MPEG-B Part 5), describes the bitstream syntax and the parsing rules. A pertinent subset of this BSDL named RVC-BSDL is defined within the scope of the current RVC framework. This RVC-BSDL also includes possibilities for further extensions, which are necessary to provide complete description of video bitstreams. RVC-BSDL specified normatively within the scope of the RVC framework is provided in ISO/IEC 23001-4.

The decoder configuration specified using FNL, together with the specification of the bitstream syntax using RVC-BSDL fully specifies the ADM and provides an “executable” model of the RVC decoder description.

The instantiated ADM includes the information about the selected FUs and how they should be connected. As already mentioned, the FND with the network connection information is expressed by using FNL. Furthermore, the RVC framework specifies and uses a dataflow-oriented language called RVC-CAL for describing FUs' behavior. The normative specification of RVC-CAL is provided in ISO/IEC 23001-4. The ADM is the behavioral model that should be referred to in order to implement any RVC conformant decoder. Any RVC compliant

decoding solution/implementation can be achieved by using proprietary non-normative tools and mechanisms that yield decoders that behave equivalent to the RVC ADM.

The decoder description, the MPEG tool library, and the associated instantiation of an ADM are normative. More precisely, the ADM is intended to be normative in terms of a behavioral model. In other words what is normative is the input/output behavior of the complete ADM as well as the input/output behavior of all the FUs that are included in the ADM.

Information technology — MPEG video technologies —

Part 4: Video tool library

1 Scope

This part of ISO/IEC 23002 defines the description of the MPEG video tool library (VTL) based on the decoder description specified in ISO/IEC 23001-4. This tool library defines the specification of FUs, which are sufficient to build complete decoding solutions according to the following coding standards:

- ISO/IEC 14496-2 (MPEG-4 Simple Profile), and
- ISO/IEC 14496-10 (MPEG-4 AVC Constrained Baseline Profile).

The objective of ISO/IEC 23001-4 is to define the general framework principles, and this part of ISO/IEC 23002 defines the MPEG VTL that includes relevant tools (or FUs) from the existing MPEG coding standards. Each FU is defined in the form of a textual description, which can be found in 4.1. The input and output behavior follows the conventions described in Clause 5 (general-purpose FUs), Clause 6 (MPEG-4 FUs), and Clause 7 (MPEG-4 AVC FUs).

This part of ISO/IEC 23002 compliant implementations can be designed using any software or hardware language and components. The reference software for the textual specification of FUs is written in RVC-CAL language of which a formal syntax is provided in ISO/IEC 23001-4, and which will be defined in Amendment 1 to ISO/IEC 23002-4.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 14496-2, *Information technology — Coding of audio-visual objects — Part 2: Visual*

ISO/IEC 23001-4, *Information technology — MPEG systems technologies — Part 4: Codec configuration representation*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23001-4 apply.

4 FU description convention

4.1 FU interfaces

As shown in Table 1, each FU is described with the following elements:

- **FU Name**: Name to represent the functional unit in this specification. The name of the FU is normative and follows the naming convention described in Annex A.1.
- **Description**: Textual explanation to describe the functionality of the FU. The description must be concise. The precise normative behaviour of the algorithm (input/output, timing etc.) is specified by the the RVC-CAL reference code in Amendment 1.
- **Profiles@levels supported**: The profiles@level supported for this functional unit. It may append that a given range of values makes the FU behave for a given profile@level and another range of values makes the FU behave for another profile@level.
- **Input**: A token that is entering the FU through the designated input port. The token type refers to the token pool described in 4.3. The 'name' field indicates the input port.
- **Output**: A token that is coming out of the FU through the designated output port. The 'name' field indicates the output port.
- **Parameter (*optional*)**: Parameters are optionally described to adjust the behavior of the FU. All the parameters must be specified with name, description and range.

Table 1 — Template of description of an FU (example)

FU Name	e.g. Algo_IDCT2D_ISOIEC_23002_1	
Description	e.g.This module computes the 8x8 Inverse Discrete Cosine Transform (IDCT) defined as $f(x,y)=\frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{2N}\cos\frac{(2y+1)v\pi}{2N}$ with u, v, x, y = 0, 1, 2, …, N-1 where x, y are spatial coordinates in the sample domain u, v are coordinates in the transform domain $C(u),C(v)=\begin{cases}\frac{1}{\sqrt{2}} & \text{for } u,v=0\\ 1 & \text{otherwise}\end{cases}$ It inputs a list of 64 coefficients and outputs a list of 64 decoded coefficients.	
Profiles@levels supported	e.g. MPEG-4 SP	
Input		
Name	**Token**	
e.g. X	e.g. BLOCK token	
Output		
Name	**Token**	
e.g. Y	e.g. BLOCK token	
Parameter		
Name	**Description**	**Range**

4.2 FU IDs

FU of the specific functionality is identified by its unique identification number. **Table 2** lists IDs and names of all FUs in VTL. IDs and names are used in FND to select FUs.

Table 2 — List of FUs and their IDs

ID	FU Name
1	org.sc29.wg11.common.Algo_ SynP_Generic
2	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_ SynP
3	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_BlockExpand
4	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420B
5	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420MV
6	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_MVR_MedianOfThreeLeftAndTopAndTopRight
7	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_MVSequence_LeftAndTopAndTopRight
9	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter_420_TYPE
10	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB6_MPEG4Part2
11	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB7_MPEG4Part2
12	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB8_MPEG4Part2
13	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB12_MPEG4Part2
14	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB13_MPEG4Part2
15	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB14_MPEG4Part2
16	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB15_MPEG4Part2
17	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB16_MPEG4Part2
18	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB17_MPEG4Part2
19	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IQ_QSAndQmatrixMp4vOrH263Scaler
20	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAddr_ThreeLeftTop_8x8
21	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAddr_ThreeLeftTop_16x16
22	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_CHROMA_8x8
23	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_LUMA_16x16
24	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8
25	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8
26	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_16x16
27	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IDCT2D_ISOIEC_23002_1
28	org.sc29.wg11.mpeg4.part2.sp.texture.Mgnt_DCSplit
29	org.sc29.wg11.mpeg4.part2.sp.motion.Mgnt_FB_w_Addess_8x8
30	org.sc29.wg11.mpeg4.part2.sp.motion.Mgnt_FB_w_Addess_16x16
31	org.sc29.wg11.mpeg4.part2.sp.motion.Algo_PictureReconstruction_Saturation
32	org.sc29.wg11.mpeg4.part2.sp.motion.Algo_Interp_HalfpelBilinearRoundingControl
33	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_NALU
34	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_SynP
35	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_BlockExpand

ID	FU Name
36	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_BlockSplit
37	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_IntraPred_Split
38	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_Parser_I_PCM
39	org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Chroma
40	org.sc29.wg11.mpeg4.part10.cbp. selectMacroblock.Algo_DemuxParserInfoForBlocks_Luma
41	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4
42	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d
43	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4
44	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Reordering
45	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Scaling
46	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_CHROMA
47	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d
48	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift
49	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IntraPred_LUMA_16x16
50	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IntraPred_LUMA_4x4
51	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_16x16
52	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4
53	org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_INTRA16x16
54	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Merge_4x4_to_8x8
55	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA
56	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16
57	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_4x4
58	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_IQ_Chroma
59	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb
60	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_YxY
61	org.sc29.wg11.mpeg4.part10.cbp. intraPred.Algo_Split_16x16_to_4x4_norasterscan
62	org.sc29.wg11.mpeg4.part10.cbp. intraPred.Algo_Merge_4x4_to_16x16_norasterscan
63	org.sc29.wg11.mpeg4.part10.cbp. interPred.Algo_Interp_Bilinear
64	org.sc29.wg11.mpeg4.part10.cbp. interPred.Algo_Interp_SeparableSixTapQuarterPel
65	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Reord
66	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MvLXReconstr
67	org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB
68	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MMCO
69	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefIdxtoFrameNum
70	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefList
71	org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_InterPred
72	org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Mgnt_DBF_AdaptiveFilter
73	org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_DBF_AdaptiveFilter
74	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_POC

ID	FU Name
75	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_BufferRender
76	org.sc29.wg11.common.Algo_Add_PixSat
77	org.sc29.wg11.common.Mgnt_Select_MB_4
78	org.sc29.wg11.common.Algo_Byte2bit
79	org.sc29.wg11.common.Algo_Mgnt_Merger420
80	org.sc29.wg11.common. Mgnt_Select_MB_8
81	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_Merger420_AVC
82	org.sc29.wg11.mpeg4.part10.php.parser.Algo_SynP
83	org.sc29.wg11.mpeg4.part10.php.parser.Algo_BlockExpand
84	org.sc29.wg11.mpeg4.part10.php.selectMacroblock.Algo_DemuxParserInfoForBlocks_Luma
85	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IS_Zigzag_8x8
86	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_8x8
87	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IIT_8x8
88	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_IntraPred_LUMA_8x8
89	org.sc29.wg11.mpeg4.part10.php.Intrapred.Mgnt_Intra_8x8
90	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_Merge_8x8_to_16x16
91	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_CHROMA
92	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_LUMA_Scaling
93	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4
94	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_Merge_8x8_to_16x16_norasterscan
95	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_Split_16x16_to_8x8_norasterscan
96	org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_demux
97	org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_mux
98	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_GeneratePredWeight
99	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectMvpLX
100	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvLXReconstr
101	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvBuffer
102	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectMvpLX
103	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectRefIdx
104	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_FrameNumToPocList
105	org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_DBF_AdaptiveFilter
106	org.sc29.wg11.mpeg4.part10.php.deblockingFilter. Algo_MvComponentReorder

4.3 Token Pool

Every token is listed in the ‘token pool’ that is the table of managing all tokens used in VTL. To facilitate the feasibility of connections among input and output ports of different FUs described in this specification, Table 3 lists all data elements (called “token”, which is used throughout this document). The ID field here is informative and used for easy lookup.